Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-93-38

Document Recognition of Printed Scores and Transformation into MIDI

Stephan Baumann

August 1993

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

Postfach 20 80
67608 Kaiserslautern, FRG
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
66123 Saarbrücken, FRG
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-5341

Deutsches Forschungszentrum für Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern and Saarbrücken is a non-profit organization which was founded in 1988. The shareholder companies are Atlas Elektronik, Daimler-Benz, Fraunhofer Gesellschaft, GMD, IBM, Insiders, Mannesmann-Kienzle, SEMA Group, Siemens and Siemens-Nixdorf. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI conducts application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to construct *systems with technical knowledge and common sense* which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ❑ Intelligent Engineering Systems
- ❑ Intelligent User Interfaces
- ❑ Computer Linguistics
- ❑ Programming Systems
- ❑ Deduction and Multiagent Systems
- ❑ Document Analysis and Office Automation.

The DFKI strives at making its research results available to the scientific community. There exist many contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Friedrich J. Wendl
Director

Document Recognition of Printed Scores and Transformation into MIDI

Stephan Baumann

DFKI-RR-93-38

A paper version of this work has been published in "Advances in Structural and Syntactic Pattern Recognition", Proceedings of the International Workshop on Structural and Syntactic Pattern Recognition, August 1992, ed. H.Bunke, World Scientific.

This work has been supported by a grant from The Federal Ministry for Research and Technology (FKZ ITW-9003 0).

Document Recognition of Printed Scores and Transformation into MIDI

STEPHAN BAUMANN

Authors´ Abstract — The processing of printed music pieces on paper images is an interesting application to analyse printed information by a computer. The music notation presented on paper should be recognized and reproduced. Numerous methods of image processing and knowlegde-based procedures are necessary. The DOREMIDI System allows the processing of simple piano music pieces for two hands characterized by the following steps:

- Scanning paper images
- Processing of binary image data into basic components
- Knowledge-based analysis and symbolic representation of a musical score
- Visual and acoustic reproduction of the results.

DOREMIDI has been realised on a Macintosh II, using Common-Lisp (Clos) programming language. The user interface is equivalent to the common Macintosh-interface, which enables in an uncomplicated way to use windows and menus. A keyboard presents the results of the acoustical reproduction.

Keywords: optical music recognition, document analysis, image processing, musical knowledge representation.

CONTENTS:

1. INTRODUCTION

Continuous progress in computer science technology has a great influence on musical applications. It is inconceivable to make a music production today without the help of a computer; automatic musical performances by means of electronic instruments are just about reality and the definition "desktop-music-production" is not just a slogan.

A considerable work improvement due to the notation programs, offers digital editing possibility similar to text and graphic systems. In case of output media, matrix printer, laser printer and foto typesetting are at disposal. Complicated, repetitional processes, (for example: transposition and voice excerpts) can be automated with help of a stored score (partitur). Systems which subject matter on the one hand is a computer supported analysis for various musical styles, and on the other hand a computer supported composition, are the object of the current research. Therefore, increasingly, standard music data bases are necessary.

Since 1987 the ANSI X3V1.8M-Working Group is working-out an equivalent proposal for a SMDL (Standard Music Document Language). So far, one has to be content with the MIDI-Standard since an international standard for the representation of music notation has not yet been dismissed. It refers to a protocol which already was worked out in 1981 by the MIDI-Standard (Musical Instrument Digital Interface) Association [MMA 88], in cooperation with various leading manufacturers of electronic musical instruments.

A keyboard equipped with a MIDI-interface is a comfortable, practice-oriented input media for music processing systems, however, music notation contains much more information than the technically oriented MIDI-protocol permits. An automatic data acquisition which allows a printed sample of common music notation (CMN) to be transferred by a scanner and subsequent processing steps into a symbolic representation is therefore meaningful, and the object to the following presentation.

2. ELECTRONIC PROCESSING OF SHEET MUSIC

In the past few years many different systems for automatic processing of music notations have been developed. 1966 a paper (one of the first in this field) entitled "Automatic Recognition of Sheet Music" [Kassler 72] was presented by Pruslin.

Pruslin concentrates on the subject of common music notation. These are merely so-called solid-body-notes, e.g. quarter notes and smaller time values. Prohibited are: whole and half notes, rests, accidentals, treble clef, bass clef, barunit indication as well as all the extra symbols (dynamics, phrasing etc.). Another limitation is that the eighth notes and the smaller time values have to be connected by a beam. For the input media a scanner is at disposal. A musical presentation which fulfills the limitations is converted by the scanner into a bitmap. The position and the width of the stafflines are determined within the framework of preprocessing. By turning of the horizontal and the vertical lines, a subsequent segmentation into coherent black areas, so-called 'contour traces' is made possible. Each of these contour traces represents either a note cluster or a timing complex. e.g. one or more beams. Finally, further processes classify an adequate length and pitch of each note.

Because Pruslins' preprocessing destroyed most of the muscial symbols, except quarter notes and beamed notegroups, an expansion of his methods for recognition of all symbols in common music notation, was not possible. Therefore, Prerau chose another method, described in his thesis "Computer Pattern Recognition of Standard Engraved Music Notation" [Prerau 70, 71, 75]. His system DO-RE-MI mainly consists of three parts: Input Section, Isolation Section and Recognition Section. Within the Input Section an conversion takes place of two or three scanned music notation barunits into one 0/1 - Bitmap. For reasons of economical storage location they are transmitted in a packed form to the Isolation Section. The Isolation Section isolates each musical symbol found in the Bitmap. The fact that the staff graphically joins the symbols, complicates this process. Above all, the method developed by Prerau, determines fragments, e.g. coherent black areas between the adjacent stafflines and connects the adjacent fragments, according to certain rules, into components, which correspond to musical symbols. The definite allocation of a component belonging to musical symbol takes place in the Recognition Section. A typical feature of a muscial symbol is the significant difference in the relation between height and width; for example, a dot is very small, the accidental Be relatively thin, a beamed notegroup very wide, etc., DO-RE-MI uses this feature to allocate the unknown components in reference to their height and width of musical symbols alternatives in question (typically 3 to 5). Using further rules which represent an algorithmic description of musical notation syntax (for example "an accidental is written on the left side of a note"), will restrict the number of possible alternatives to a definite hypothesis. As soon as all components are recognized, the program produces an output in Ford-Columbia-Representation M. R. L. This Music Representation Language by Stefan Bauer-Mengelberg [BauerMengelberg 70] was developed 1970 and does not anymore correspond with the standards of musical applications. However, the idea of rough classification of different height and width relations is very important to the musical symbol recognition.

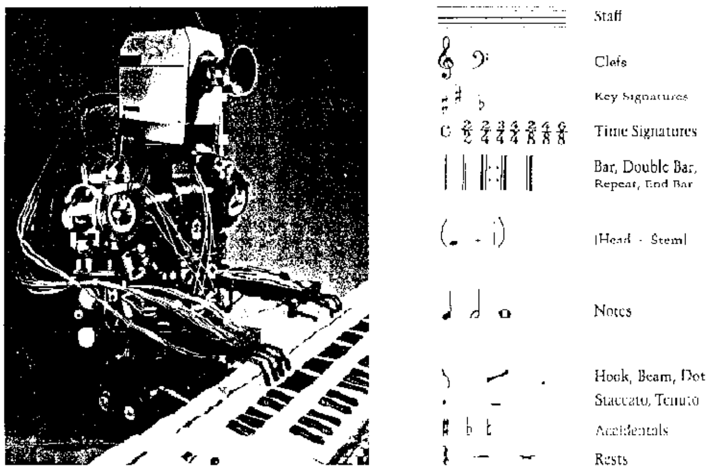


Fig. 1: WABOT-2 and recognized musical symbols [Matsushima 85]

The two papers mentioned above were developed without considering the processing speed. This fact was an essential factor for a project at the University Waseda in Tokyo

[Matsushima 85]. During the development of the WABOT-2 robot, a High-Speed-Recognition system was used, which allows the real-time music performance. Figure 1 shows the roboter and the recognized music notation. The relevant symbols, such as stafflines, head, stem are recognized via hardware, through which it is possible to recognize a simple piece of music within an extremely short time (approximately 10 seconds). However, the system is not able to recognize complex music notations. The roboter only plays simple nursery songs on an organ. Hereby three staff systems are recognized. The top system contains information for the right hand, the central system for the left hand and the low one serves as an information source for the foot. The deployment of hardware-based procedures to Template Matching, explains the fast processing time. However, when this procedure is confronted with complex scores, it leads immediately to incorrect results. Therefore, this system is very efficient in its restricted range of application, however it cannot be universally applicable.

A very promissing approach is described by Kato and Inokuchi [Kato et al. 90]. The Recognition System for Printed Piano Music Using Musical Knowledge and Constraints, is capable of recognizing almost all symbols of CMN (Common Music Notation), in simple piano pieces, with an accuracy of almost 90%. The basic idea is a Top-Down-Approach, where an effective, knowledge-based recognition within a barunit is made possible. The few overlapping barunit symbols, such as ties, are processed separately in a rework phase. In the preprocessing phase the width and distance of the stafflines is determined. The next step determines the positions of single stafflines and bars. This way the segmentation into barunits is reached, and a bar-unit-recognition-phase for each barunit can be developed. This phase contains: elimination of stafflines, the recognition of attributed symbols (clefs, time signature, key signature) as well as the actual note symbol recognition. Basically it is accomplished by four preprocessing phases (primitive extraction, symbol synthesis, symbol recognition, semantic analysis). These phases work in a working memory structured into five levels (image, primitive, symbol, meaning, goal). Only the lowest image level in the initial phase has a content. Through the application of processing modules, the working memory is being modified until a generated hypothesis on the highest level represents the results of recognition process. Finally, in the postprocessing phase the barunit-overlapping global symbols are recognized, and the results of all barunits accordingly modified and put together to a total result. Kato's and Inokuchi's system is capable to handle simple as well as complex notations, where high quality printing copy brings fast results, and a poor image quality enforces time consuming analyses. Figure 2 shows examplary some of the test results of the system, whereby the recognition rate of the analyzed piece decreases with the increasing complexity.

Piece	*Recognition rate*
Für Elise (L.V. Beethoven Op.178)	95.6 %
Türkischer Marsch (W.A. Mozart)	91.5 %
Etude (F.Chopin Op. 10 Nr 3)	87.1 %
Sonata Pathetic (L.V.Beethoven)	83.3 %

Fig.2: Recognition Rates of the Kato&Inokuchi-System[Kato et al. 90]

The excellent test results of this attempt speak in favour of the top-down-oriented processing relying on the barunits. Therefore, this paper leans considerably on this approach. In order to develop a prototypical recognition system for printed music notation in a restricted amount of time, a delimitation was chosen in comparison to the introduced systems:

Restrictions:

- limitated to two-handed piano pieces with monophonic voices
- usage of existing methods for image analysis, which have been developed in the ALV-project (Automatisches Lesen und Verstehen)

Features:

- extension through modular program design
- processing of a complete page (DIN A4) of a score
- generation of MIDI-Standard-File as output format and interface for existing notation and sequencer software.
- acoustical interpretation of the obtained results.

3.THE DOREMIDI SYSTEM

The DoReMIDI system (DOcument REcognition of Printed Scores and MIDI-Generation) for automatic processing of music pieces consists of many single phases. These are scematically presented in Figure 3.

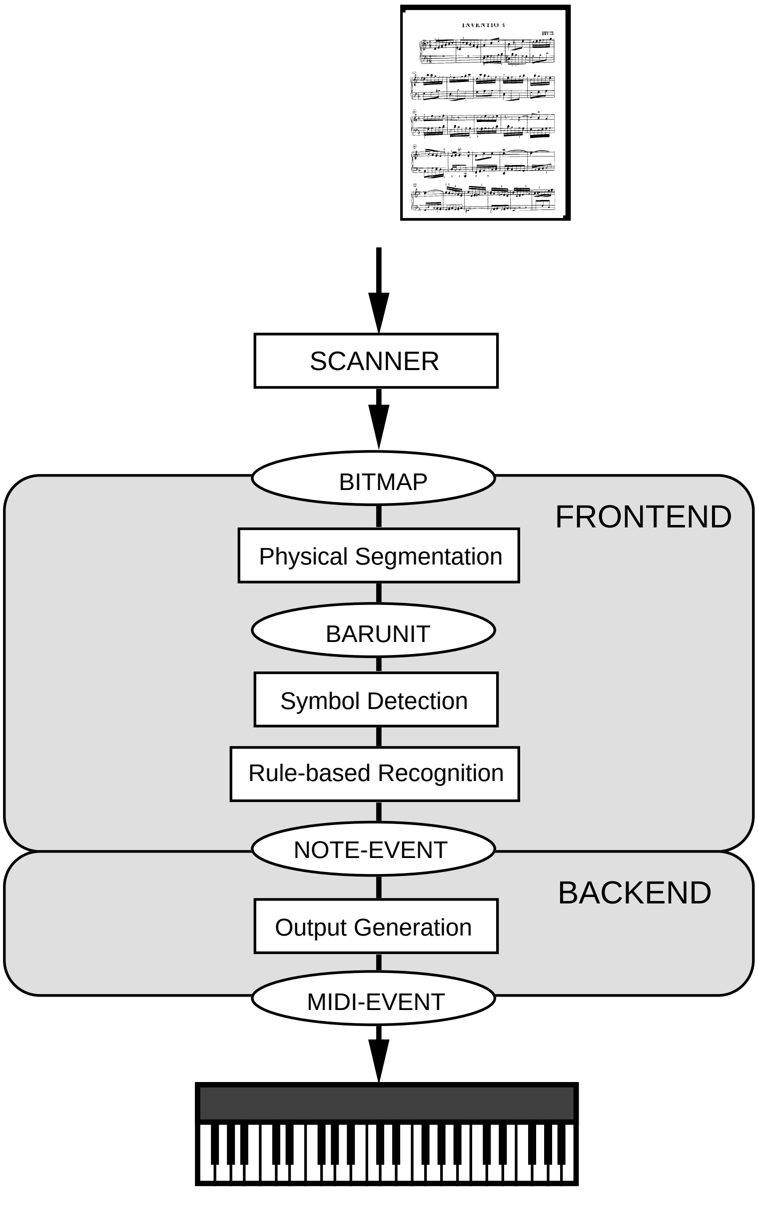


Fig.3: The DOREMIDI-System

To beginn with, a music notation printed on a piece of paper (a simple two-handed piano piece) will be scanned by using a 200dpi (dots per inch) resolution, equivalent to faximile quality. This binary picture is subdivided into physical basic segments, the single barunits of the piece. The following phase 'symbol detection' and 'rule-based recognition' operate on the following basis. Stafflines within a barunit are removed without fragmenting the symbols. The actual symbol detection, e.g. the description of musical objects within a barunit by appropriate data structure, is performed by employment of puzzle trees [Dengel 92]. Using this technique an adequate basis is given, in order to handle the central problem – the knowledge based symbol recognition– in a meaningful way. This is made possible, because the significant differences in relation between height and width of musical symbols can be used for a primary classification of a component found in a barunit. The circumscribing rectangles for each symbol are extracted from the puzzle tree representation. Since the puzzle tree further on represents the complete object description on a pixel level, other analyses on the same data structure can be produced. A definite conclusion about the significance of different symbols in a music piece is reached by the deployment of a music rule base. Essentially the symbol recognition is completed, and now a symbolic representation of a music piece can be generated. Incorporating barunits overlapping context information such as, ties and accidentals, intermediate-format NOTE-EVENT is produced. This represents the interface between Frontend and Backend. Frontend includes all phases needed for symbol recognition. Backend includes those phases which are used for generating an output format for producing an acoustical interpretation. From the intermediate-format in the backend, step by step so called MIDI-EVENTs are produced, and stored as a MIDI-Standard-File. This file serves as an input for a commercial sequencer software (EZ-Vision of Opcode Systems), which triggers a keyboard linked with MIDI-Interface to realize the acoustical interpretation of the results. A visualization of the recognized music notation based on MIDI-Standard-Files is posssible with any common notation software (for example ProComposer by Mark of the Unicorn). These only shortly described processing steps will be handled more detailed in chapter 3.1 'Preprocessing', 3.2 'Barunits as Segmentation Units', 3.3 'Knowledge-based Symbol Recognition' and 3.4 'Output Generation'.

3.1. PREPROCESSING

(A) Hierarchical Score Layout

The documents (printed partiturs) to be analyzed are simple piano pieces for two hands. A typical construction of a DIN A4 page is shown in Figure 4. The distinct hierarchical construction of the partitur supports a top-down procedure and should be considered by the formal representation in an actual implementation. Therefore, an object-oriented representation form was selected, so that hierarchical relations in an implementation can easily be formed. Structures resulting from segmentation can be classified – with help of this approach – into different object classes related with each other by a definite hierarchical relation. Furthermore, this allows inheritance of common variables. The definition of class-specific operations as methods completes the priniciple of data encapsulation. The representation of the processed score occurs in run-time-generated instances of the object classes shown in Figure 5.

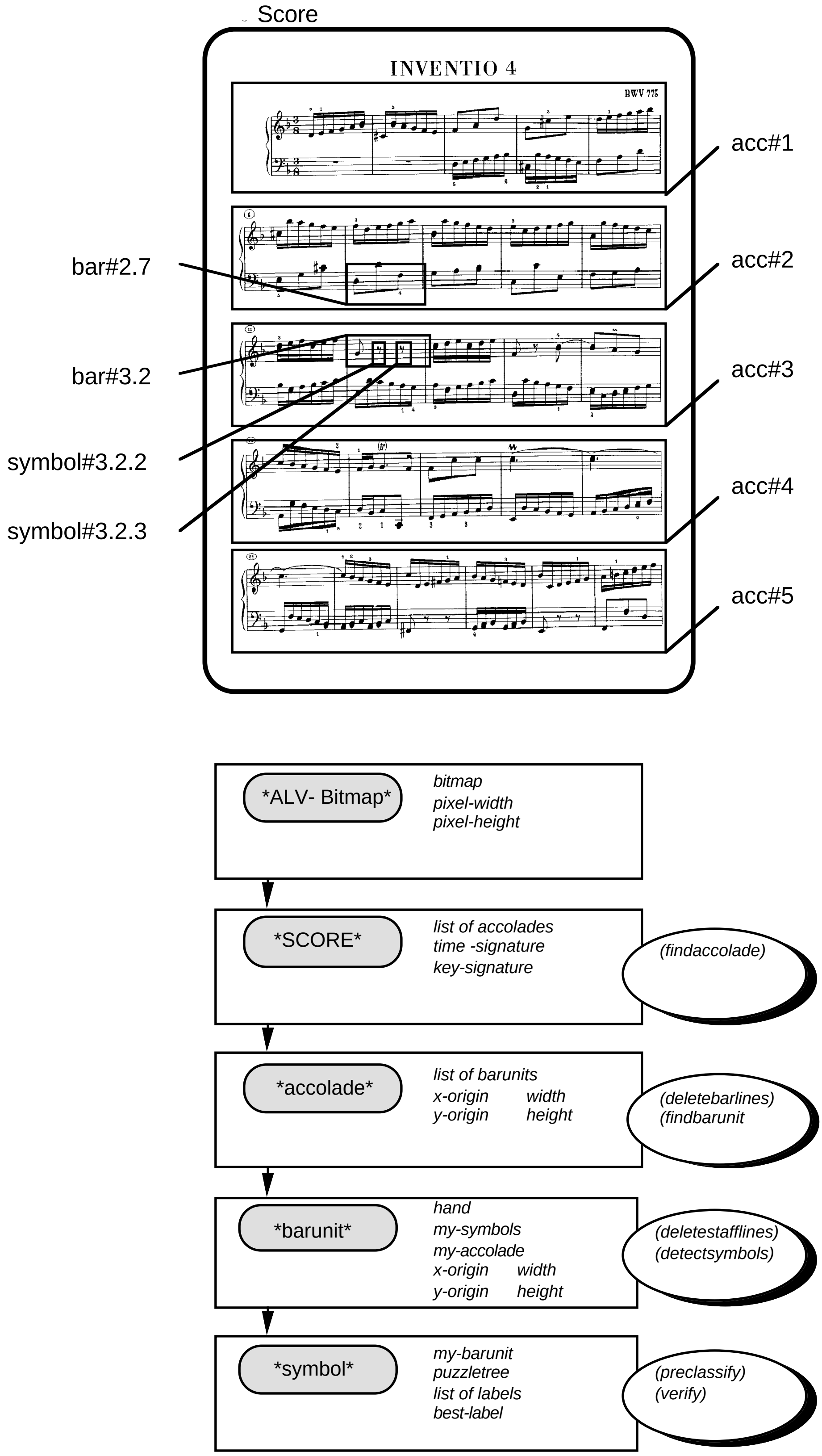


Fig.5: Score Layout & Object Class-Hierarchy

(B) Segmentation

The object of segmentation is generating the instances of object classes presented in (A); and covering the corresponding data slots with values. The segmentation occurs in 2 steps. Since the input notes are for piano pieces for two hands, respectively two staff systems are needed. Here the upper system serves the right hand, the lower system contains information for the left hand. Both systems are combined to a 'accolade' by a accolade bracket. The unbroken barlines in the accolade are arranged vertically. Several barunits exist in this accolade class instance. In order to find them it is necessary to remove the unbroken barlines. This occurs by using the method 'delete barlines' for an accolade. Subsequently the barunits are discovered and supervised in each accolade. The results of the overall procedure is shown in Figure 6.

Fig.6: Accolade: (a) raw data, (b) barline removal, (c) resulting barunits

The symbol recognition procedure and musical analyses described in the following chapter, are performed only in the barunits detected in the preprocessing.

3.2 BARUNITS AS SEGMENTATION UNITS

Single barunits in a music piece can be interpreted as physical fundamental segments. They contain the musical symbols to be identified, and represent a closed unit of the musical rule system. Because the individual symbols in a barunit are connected together with stafflines, and so represent a coherent image structure, the stafflines have to be removed at the beginning. Afterwards, the isolated symbols can be grouped and represented through adequate data structure.

(A) Staffline Removal

The removal of staff lines has basically two main problems:

1. The location of the staff has to be exactly determined;

2. Deleting of the staff lines should avoid:

- remaining stafflines which could be mistaken for musical symbols

- removal of image information belonging to musical symbols.

If these two important aspects are ignored, much more effort in the next phase (symbol recognition) for correct recognition of the existing information will be necessary. There is a proportional trade-off between the correct procedure for staff removal and the amount of computation for the detection and classification procedures of musical symbols. The following figure 7 shows the results of this phase.

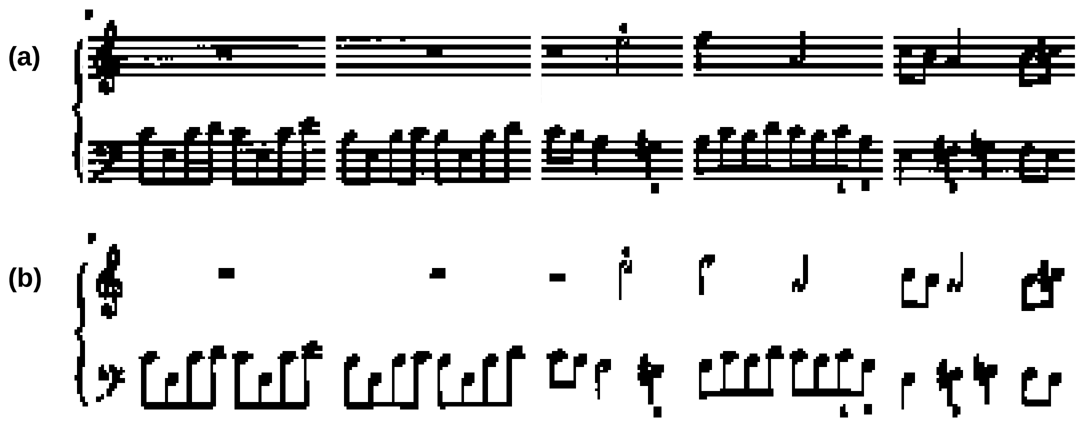


Fig.7: Accolade: (a) presegmented data, (b) staffline removal

(B) Symbol Detection by Use of Puzzletrees

After the staff lines are removed from all barunits, without fragmenting the musical symbols, a suitable representation for the remaining image data has to be found.

The puzzletree [Dengel 92] is an appropriate way for the representation, because it is a technique which accepts for the input rectangle pictures in all sizes and with successive decomposition disassembles a minimal amount of rectanglular blocks. These blocks devide the given image area into regions with pixels of the same colour; in this case black or white. When staffs are removed, barunits serve as output for the puzzletree. We speak of rectangular image formats which contain the connected black and white areas (in this case the musical symbols) and therefore, they fulfill the input requirements for puzzletree technique application.The application of puzzletree technique delivers a number of image blocks which reconstruct the original image just like a puzzle would. This technique is applied in a single barunit as long as only white blocks are obtained. In case this criteria cannot be fulfilled, the decomposition will stop and deliver the circumscribing rectangles of the contained musical symbols. The results of this procedure are shown in Figure 8.

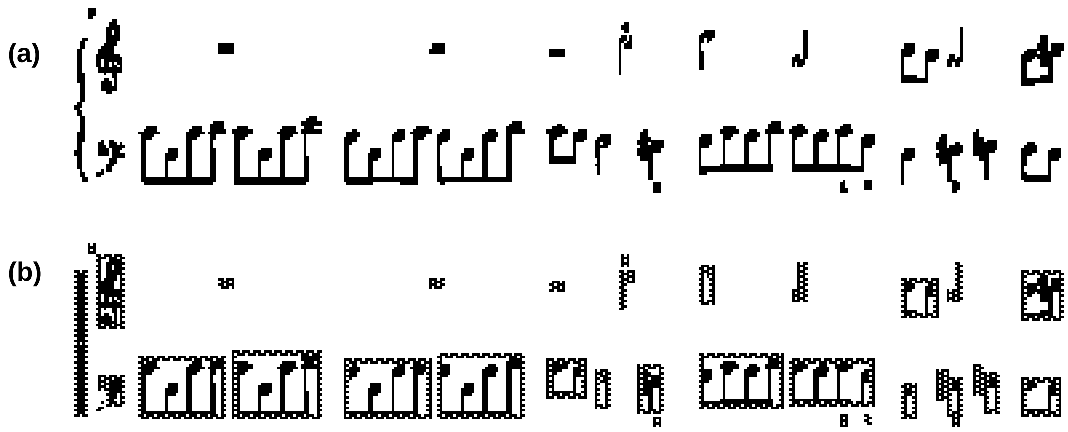


Fig.8: Accolade: (a) input data, (b) detection of symbols

3.3. KNOWLEDGE-BASED SYMBOL RECOGNITION

(A) Rough Classification of Musical Objects

When the symbol detection described in chapter 3.2.(B) is completed, a puzzletree and a circumscribing rectangle description is given for each symbol in the partitur. With this information a rough classification to allow symbol recognition is possible. The subset of musical symbols consits only of symobls which can be arranged into those classes shown in Figure 9.

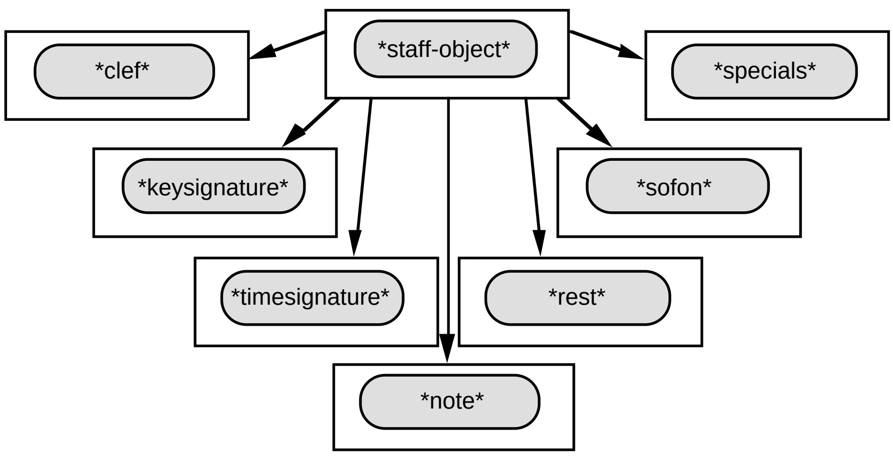


Fig.9: Classes of musical symbols

Musical symbols significantly differ in their height and width relationship. This feature can be very helpfull when by means of measurements of the circumscribing rectangle, the assignment of an object to a class is made. In order to classify the discovered symbols of a barunit to a class, a decision tree classifier is used [Dengel 89]. The decision tree is a simple technique to reduce the given global decisions into numerous simple decisions. A decision tree classifier reaches a decision faster than a single-stage classifier.In the described system, a binary tree, in which nodes comparison operations take place, was chosen as a decision tree. Hereby the height or/and width of each symbol is compared with the value of the classifier. With approximately three or four comparisons a leaf of the tree is reached. These leafs represent the possible musical classes, and regulate the allocation of the symbols into a class and to the matching hypotheses. The produced possibilities for each discovered symbol, are listed in a 'list of labels'. How a decision tree classifier is constructed is shown in Figure 10.

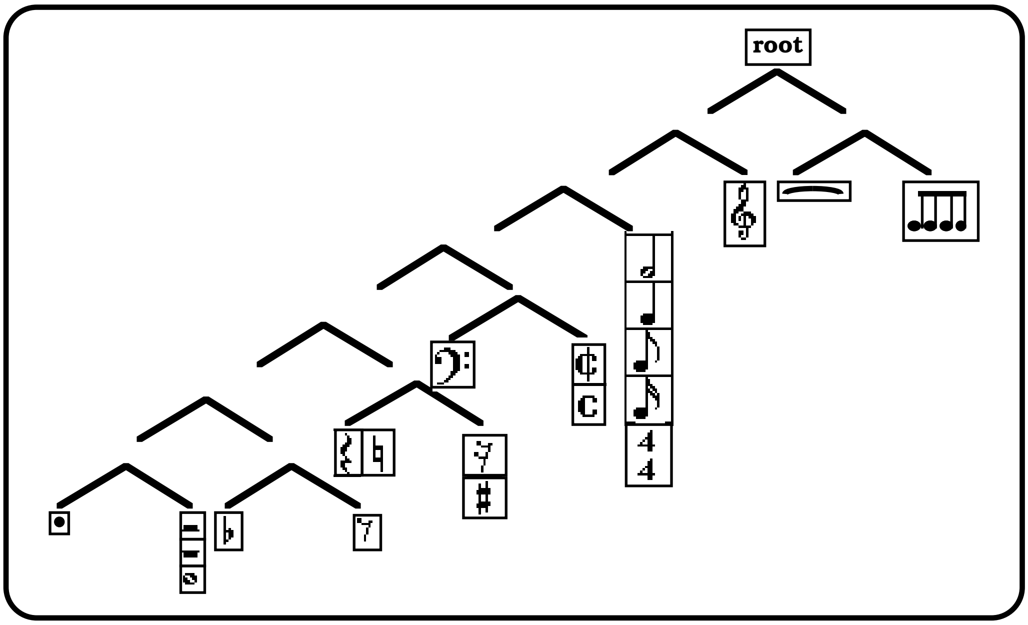


Fig.10: Decision-Tree

To reach a unique hypothesis, the following musical analysis has to be processed.

(B) Features and Musical Rules

The entries in the list of labels provide hypotheses for the possible meaning of each symbol. In order to verify them, additional knowledge is necessary.

A thorough inquiry of typical musical object features occuring in the partitur has been executed. The results of the inquiry as well as the retrospective on the theoretical musical literature (mainly [Ziegenrücker 88], [Wanske 84], [Michels 77]) are expressed in the used rules (for further details see [Baumann 92]).

3.4. OUTPUT GENERATION

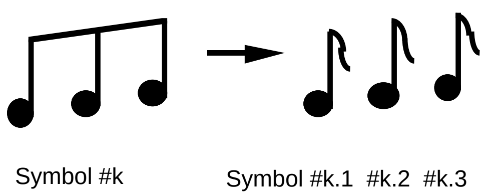
After the completion of symbol recognition and musical analysis, a specific meaning for each discovered symbol is given. In order to reach the output format, other processing steps are necessary. First an intermediate-format containing the so-called NOTE-EVENT-sequences for the left and right hand is produced, which represent the note information in a symbolical form. The MIDI-Event-Generation fulfills the requirements to generate an international standard-format for the communication between electronical music instruments , and so the desired acoustical keyboard interpretation can be realized. The methods for intermediate-format production are described in detail in Chapter (A). The preparation of acoustical interpretation is precisely described in Chapter (B) and (C).

(A) Intermediate Format NOTE-EVENT

The intermediate-format provides separately for the left and the right hand a sequence of so-called NOTE-Events. Only two informations are necessary for characterization of a single NOTE-Event: the indication of pitch and duration.

Treatment of beamed notegroups

Beamed notegroups represent a combination of notes having the same duration. This procedure helps to read the notes easier. A decomposition of such notegroups is unavoidable, because for generating an intermediate -, respectively output format individual note results have to be represented. Figure 11 shows the basic algorithm how to separate a group.




```
(split)

if label(symbol #k) = beamedNotegroup then
   for i:=1 to numbernotes
     do begin
          duration(generate(symbol #k.i):=duration(symbol#k)
          position(symbol #k.i):=position(extract(symbol#k/i)
         label(symbol #k.i):= *note*
   endfor
endif
```

Fig.11: Decomposition of a beamed notegroup

It is necessary to have the information about a note value which can be extracted from the beam, as well as the information about the pitch of the existing notes which result from the note-head-position relative to the staff system.

Employment of Contextual Information

To determine the correct length of a Note-Event ties and punctuation have to be considered. To determine the pitch the accidentals preceeding a note (sofon) as well as the keysignature and preceeding accidentals in the same bar have to be taken into account. The resulting underlying syntactic structure (under the assumption of the restricted sequential order) is shown in figure 12 .

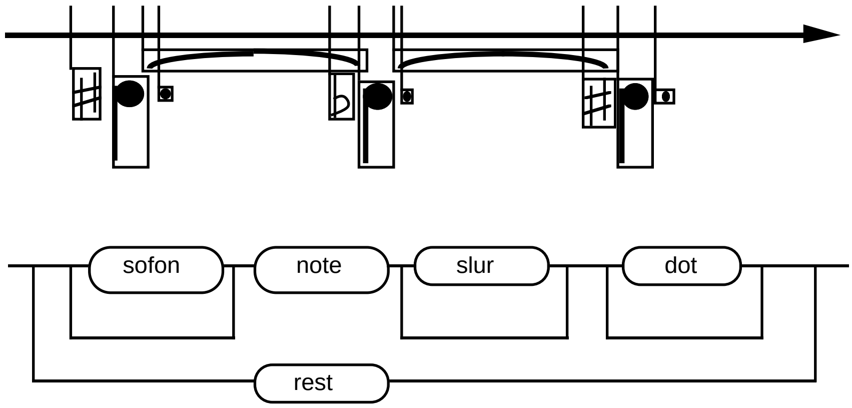


Fig.12: Syntax diagram

(B) Generation of MIDI-EVENTs

Having the intermediate NOTE-Event format allows an straightforward conversion to the desired output to trigger an external tone generator via the serial communication port. The MIDI output is an international standard which was first described in 1980 and is nowadays well established.

A single MIDI-Event consists of a three-byte packet describing a note-off or note-on-signal and the corresponding pitch and velocity of the desired note. According to the absence of information about the velocity the third byte is not relevant for this application and therefore set to a default value. Transforming the NOTE-Events into the corresponding MIDI-Events is followed by the time-specific storage of them in a so-called MIDI-Eventlist.

(C) MIDI-Standard-File

The usage of the MIDI-Standard-File (consisting of MIDI-Events and header information) as output of the DoReMIDI-system is obvious because a variety of existing musical applications are using this format as an interchange format. Thus, the described system can be used as a frontend to these applications. In our case the DoReMIDI-system generates a MIDI-file which is exported to a commercial sequencer software triggering the attached keyboard.

4. DOREMIDI - USER INTERFACE

The DoReMIDI-System has been implemented on a Macintosh II fx using Allegro Common Lisp and its objectoriented expansion. A DIN A 4 Apple scanner is used as input medium to convert a single page of printed musical score to a binary image. For the purpose of acoustical reproduction of the obtained results a keyboard is used as an external tone generator. The connection between the Macintosh and the keyboard is realized through a MIDI-Interface which is attached to the serial port of the Macintosh as shown in figure 13.

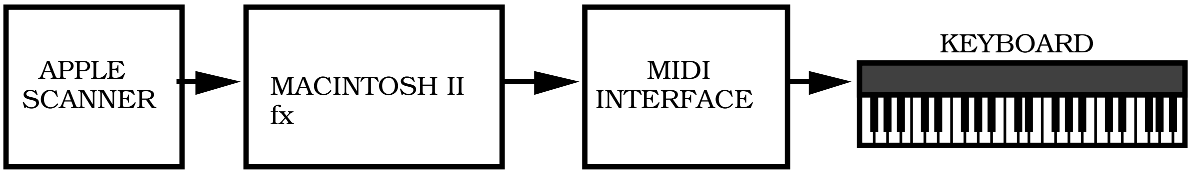


Fig.13: DoReMIDI:Hardware

The user interface follows the Macintosh Interface Guidelines which results in a typical menu-driven command-mode.

4.1. PROCESSING OF A SINGLE PAGE

Various menus and submenus represent the corresponding processing steps of the system. An incremental processing from the binary image to the MIDI-ouput is therefore given (see figs. 15-18)

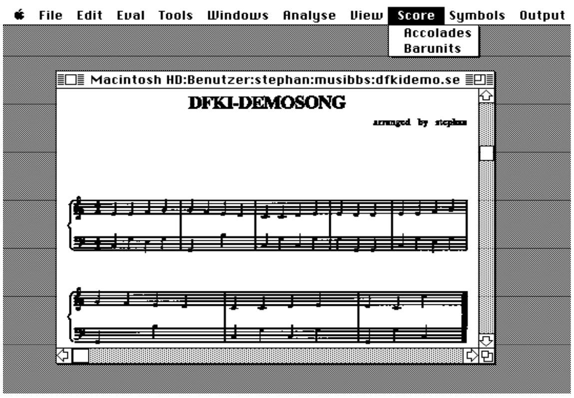


Fig.15: Segmentation

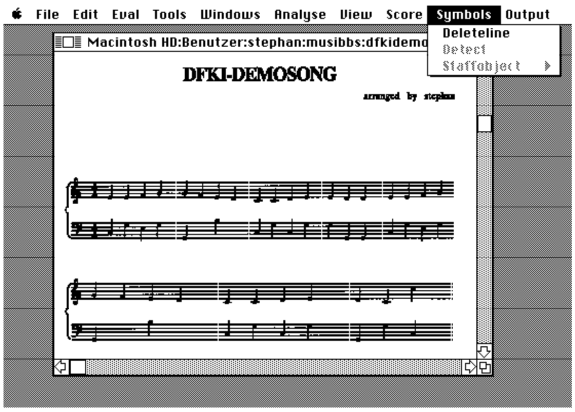


Fig.16: Staff removal

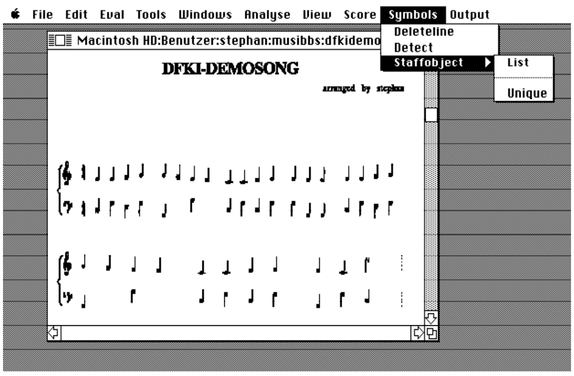


Fig.17: Symbolrecognition

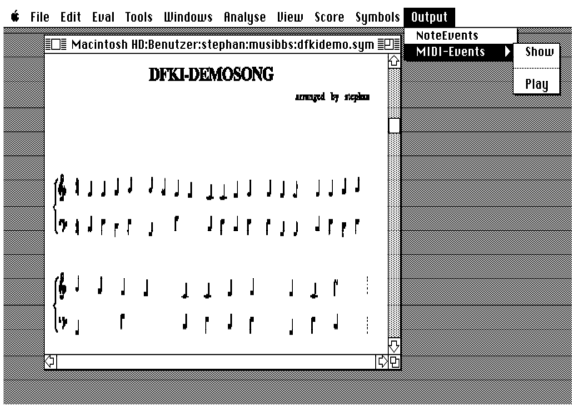


Fig.18: Generation of Output

The following short list of menu-descriptions explains the various processing steps:

-	FILE	-> NEWSCAN:	starts the scanning procedure (resolution = 200 dpi)
-	SCORE	-> ACCOLADES:	detection of the accolades
		-> BARUNITS:	deletion of barlines and subsequent detection of barunits
-	SYMBOLs	-> DELETELINE:	staffline removal
		-> DETECT:	detection of musical symbols (circumscribing rectangles)
		-> STAFFOBJECT:	-> LIST: generation of hypotheses by means of decision-tree classification
			-> UNIQUE: disambiguation to unique results
-	OUTPUT	-> NOTEEVENTS:	generation of the intermediate format
		-> MIDIEVENTS:	-> SHOW: display of the generated Midievents
			-> PLAY: save the Midievents as Standard MIDIFile
-	VIEW	->	various self-explaining menus for the visualisation of the intermediate results.

4.2. ACOUSTIC REPRODUCTION

The acoustic reproduction is based on the usage of a commercial sequencer software which offers the option to import the generated MIDI-Standardfile. We chose the EZ-Vision product because its very easy to use. Opening the submenu IMPORT under the ABLAGE-menu allows the import of the MIDIfile. The entire EZ-Vision user-interface resembles a traditional recording tape providing the user with PLAY, REWIND, RECORD, etc. Pushing the PLAY knob starts the acoustic reproduction of the generated results.

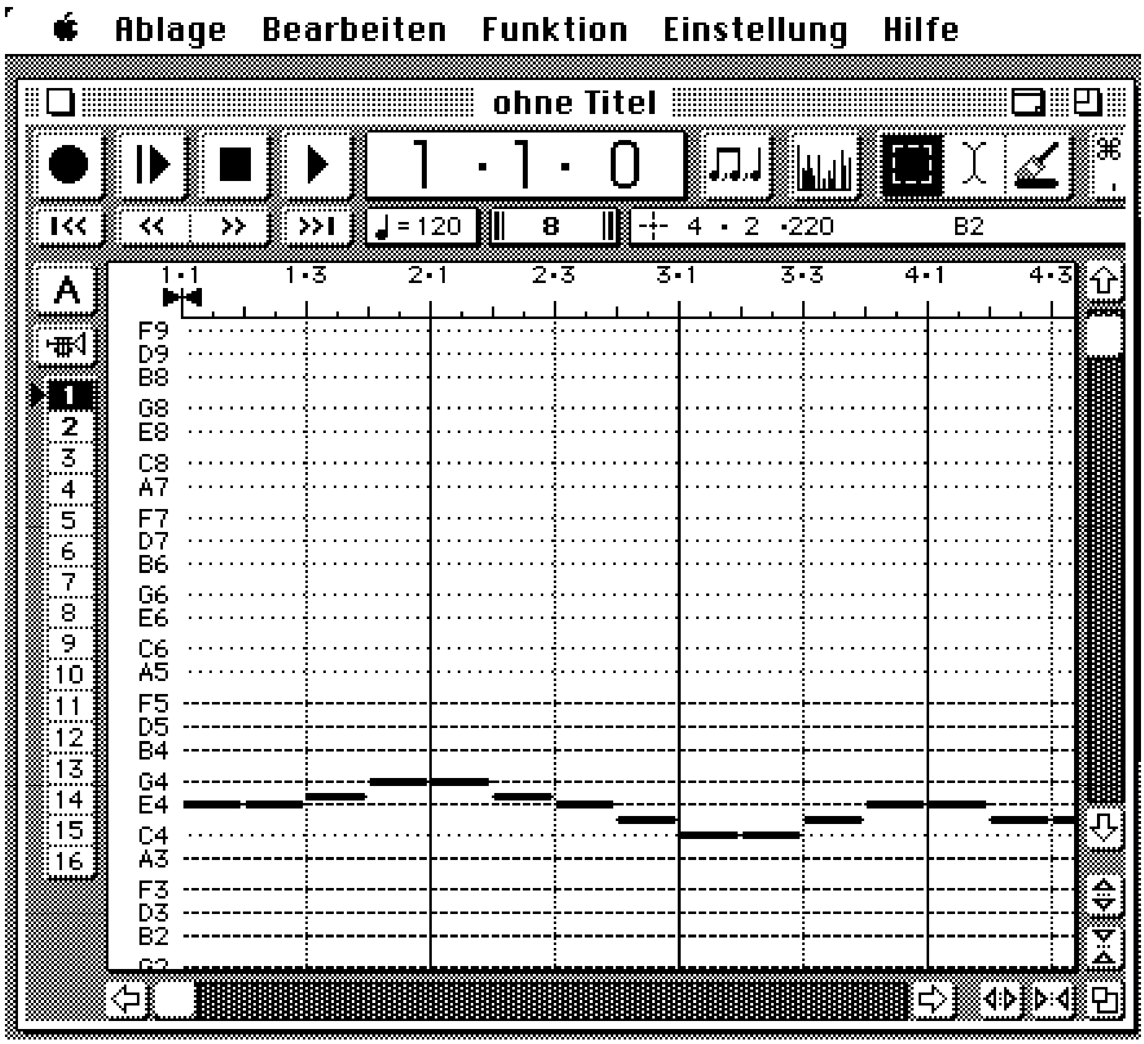


Fig.18: EZ-Vision

5. DISCUSSION

The system DoReMIDI is result of first research activities in optical music recognition. It is an experimental system focussing on a combination of methods for a complete throughput from paper to acoustic generation. It has been implemented on a McIntosh IIfx in Common Lisp. DoReMIDI is restricted on simple twohanded pieces of piano music scanned by a resolution of 200 dpi and considering a reduced set of musical symbols as well as monophony in both voices.

For testing the system, we have considered pieces of printed music of different musical complexity, ranging from professional composer output to J.S.Bach piano music. The recognition rate ranges between 80 and 100% mostly depending on the type of symbol. Main problems result from unexpected fragmentation of musical symbols because they tangentially touch a staffline. As a consequence, the fast, but simple algorithm for staffline deletion considers these symbol parts as pictoral noise and therefore, removes them. Typical examples for this erroneous behaviour are half and wholenotes filling out staffspaces. Similiar problems arise with flats and sharps touching stafflines . To handle these problems in future work, a decision has to be taken between better staffline removal or techniques for assembling symbol fragments.

Furthermore the underlying grammar has to be enriched to allow the processing of complex notations without considering certain assumptions, such as cases in which clef or key signature may change.

Because of object-oriented implementation of the system, an adaption to more complex sets of symbols can be done. For that purpose, the decision tree classifier has to be enhanced and appropriate rules have to be added to the rule base. The adaption to musical scores where accolades consist of more than two staff line systems can be done by few modifications in physical segmentation. Moreover, the tackling of polyphonic input seems to be possible by adding new techniques for isolating individual symbols.

Generating the output in terms of MIDI-EVENTs, the result produced by our system is compatible to nearly all musical devices and applications. It therefore represents a paper-to-MIDI interface.

RESULTS

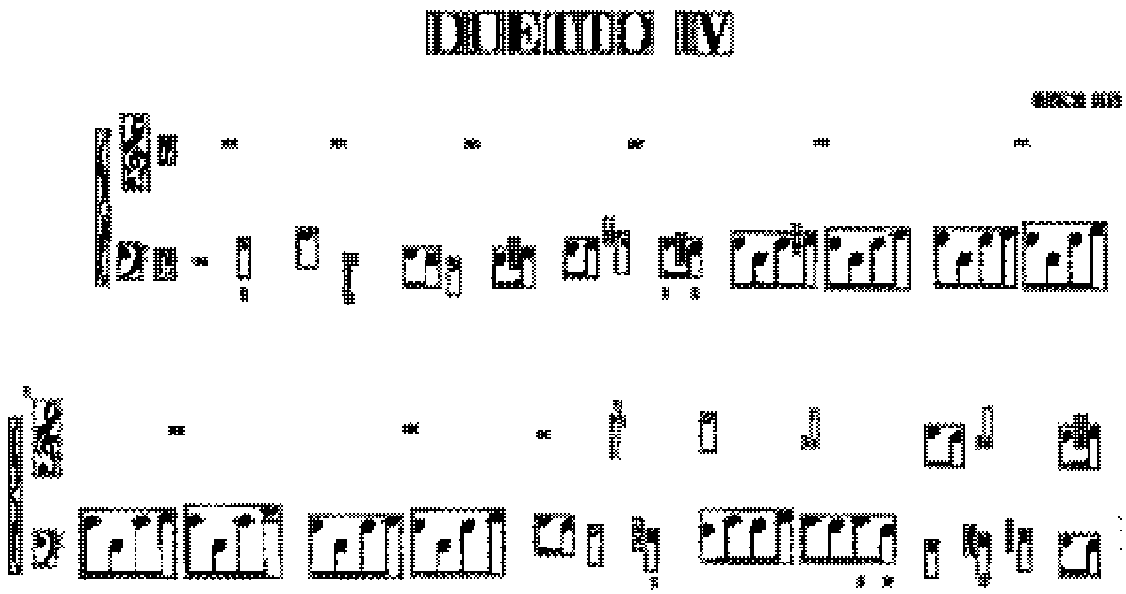
The following screenshots should give an impression of the processing phases of the DoReMIDI-System. The input was taken from [Bach 78].

(A) Input: *J.S. Bach: Duetto IV (extract)*

(B) Accolades

(C) Barunits after Barline-Deletion

(D) Symbols after Staffline Removal



(E) NoteEvents after Decision-Tree-Classification & Musical Analysis

"right-hand:"	"left-hand:"			
nil / 1	nil / 1/2	b / 1/8	c1 / 1/8	c1 / 1/8
nil / 1	a / 1/2	a / 1/8	e / 1/8	b / 1/8
nil / 1	c1 / 1/2	e / 1/8	c1 / 1/8	d1 / 1/8
nil / 1	e / 1/2	a / 1/8	e1 / 1/8	c1 / 1/8
nil / 1	f / 1/8	c1 / 1/8	b / 1/8	b / 1/8
nil / 1	e / 1/8	b / 1/8	e / 1/8	c1 / 1/8
nil / 1	d / 1/2	e / 1/8	b / 1/8	a / 1/8
nil / 1	e / 1/8	b / 1/8	c1 / 1/8	e / 1/4
nil / 1/2	f# / 1/8	c1 / 1/8	b / 1/8	f# / 1/4
e2 / 1/2	g / 1/8	b / 1/8	e / 1/8	g / 1/4
f2 / 1/2	a / 1/8	e / 1/8	b / 1/8	f / 1/8
a2 / 1/2	a# / 1/2	b / 1/8	d1 / 1/8	e / 1/8
c2 / 1/8	a / 1/8	d1 / 1/8	c1 / 1/8	
b1 / 1/8	g# / 1/8	c1 / 1/8	b / 1/8	
a1 / 1/2	a / 1/8	e / 1/8	a / 1/2	
b1 / 1/8	e / 1/8	c1 / 1/8	g# / 1/4	
c2# / 1/8	a / 1/8	d1 / 1/8	a / 1/8	

REFERENCES

[Adolph et al. 91] H. G. ADOLPH, C. MEYER: MIDI - Manie. In c´t 1/91, pp142-146.

[Bach 74] J.S.BACH: Das Wohltemperierte Klavier, Teil1.G.Henle Verlag 1974.

[Bach 78] J.S.BACH: Vier Duette.G.Henle Verlag 1974.

[Bach 78] J.S.BACH: Inventionen und Sinfonien.G.Henle Verlag 1974.

[Baggi 91] D.L.BAGGI: Computer- Generated Music. In Computer, July 1991.

[BlBaird 91] D.BLOSTEIN und H.BAIRD: A Critical Survey of Music Image Analysis. In Standard Document Analysis, Eds. H.Baird, H.Bunke and K.Yamamoto,Springer Verlag 1992.

[BauerMengelberg 70] S. BAUER-MENGELBERG: The Ford-Columbia Language.In Brooks 1970a, pp 48-52.

[Baumann 92] S. BAUMANN: Automatische Verarbeitung gedruckter Musikstücke. Diplomarbeit am DFKI-Kaiserslautern, Juni 1992.

[Baumann & Dengel 92] S. BAUMANN & A. DENGEL: Transforming Printed Piano Music into MIDI, Int'l Workshop on Syntactic and Structural Pattern Recognition, Bern, August 1992.

[Böcker et al. 90] H. D. BÖCKER, A. MAHLING und R. WEHINGER: Beyond MIDI:Knowledge - Based Support for Computer Aided Composition. In Proceedings of the International Computer Music Conference 1990, pp 284-287, Glasgow.

[Carter et al. 87] N.P.CARTER, R.A.BACON und T. MESSENGER: The Acquisition, Representation and Reconstruction of Printed Music by Computer: A Review. In Computer and the Humanities 22, pp 117-136 (1988).

[Clarke et al. 89] A.T.CLARKE, B. M. BROWN und M. P. THORNE: Using a Micro to Automate Data Acquisition in Music Publishing. In Microprocessing and Microprogramming 24, pp 549-554 (1988).

[Craner 91] P.M.CRANER: New Tool for an Ancient Art:The Computer and Music.In Computer and the Humanities 25, pp 303-313 (1991).

[Dengel 89] A. DENGEL: Automatische Visuelle Klassifikation von Dokumenten, Doktorarbeit, Fakultät Informatik der Universität Stuttgart, 1989.

[Dengel 91] A. DENGEL: An Approach for Self-Adapting Structuring of Space Using Puzzletree-Encoding, Proceedings 4th Int´l Symposium on Artificial Intelligence, Cancún, Mexico, (Nov. 1991), pp. 314-320.

[Dengel 92] A. DENGEL: ANASTASIL: A System for Low-Level and High-Level Geometric Analysis of Printed Documents, in: H. Baird, H. Bunke, K. Yamamoto (eds.) Structured Document Image Analysis, Springer Publ. (1992).

[FahmyBl 91] H. FAHMY und D. BLOSTEIN: A Graph Grammar for High- Level Recognition of Music Notation. In Proceedings of the First International Conference on Document Analysis and Recognition, September 30 - October2, 1991 Saint- Malo France.

[Fischer 91] M. FISCHER: Notenspiegel: Musikalische Notation auf dem PC. In c´t 1/91, pp 134-140.

[Grover 89] J. GROVER: A Computer- Oriented Description of Music Notation, Part 1-3. Research Report, Departement of Informatics, University of Oslo, 1989.

[Kaminski 89] P. KAMINSKI: Die MIDI- Spezifikation. Ein Nachschlagewerk von Peter Kaminski. In Keyboards 1989.

[Kassler 72] M. KASSLER: Optical-Character-Recognition of Printed Music: A Review of Two Dissertations. In Perspectives on New Music 11, pp 250-254 (1972).

[Kato et al. 90] H. KATO und S. INOKUCHI: The Recognition System of Printed Piano Music using Musical Knowledge and Constraints. In Proceedings of the IAPR 1990 SSPR Workshop, Murray Hill, NJ (13-17 June, 1990).

[Keys 90] N.N.: Der geheimnisvolle Sequencerstandard:Standard-MIDI-File Format 1.0.In Keys 4/90.

[Mahoney 82] J. V. MAHONEY: Automatic Analysis of Musical Score Images; B. S. Thesis, Dept. of Computer Science and Engineering, MIT Cambridge (May 1982).

[Martin et al. 91] P. MARTIN und C. BELLISSANT: Low- Level Analysis of Music Drawing Images. In Proceedings of the First International Conference on Document Analysis and Recognition, September 30 - October 2, Saint Malo.

[Matsushima 85] T. MATSUSHIMA: Automated High Speed Recognition of Printed Music (WABOT 2 Vision System). In Proceedings of the International Conference on Advanced Robotics, pp 477-482 (1985).

[Michels 77] U. MICHELS: DTV- Atlas zur Musik, Band 1; Deutscher Taschenbuch Verlag , ISBN 3- 423- 03022- 4 (dtv).

[MMA 88] MIDI Manufactures Association:MIDI 1.0 Detailed Specification, Int´l MIDI Assoc., Los Angeles (1988).

[Newcomb 91] S.R. NEWCOMB: Standard Music Description Language complies with hypermedia standard. In Computer, July 1991, pp76-79.

[Opcode 92] OPCODE SYSTEMS: EZ-Vision -Benutzerhandbuch,deutsche Version, 1992.

[Oppenheim 88] D.OPPENHEIM: Standard MIDI File Specification.March1, 1988.

[Pogue 91] D.POGUE: First Steps in the Sequence. In MacWorld, June 1991, pp146-153.

[Pope 88] S. T. POPE: Building Smalltalk - 80 - based Computer Music Tools. In Journal of Object - Oriented - Programming, April/May 1988 pp 6-11.

[Prerau 70] D. S. PRERAU: Computer Pattern Recognition of Standard Engraved Music Notation , MIT (September 1970): Ph.D.dissertation; thesis supervisor: Murray Eden.

[Prerau 71] D. S. PRERAU: Computer Pattern Recognition of Printed Music. In Proceedings of the Fall Joint Computer Conference, Montvale, NJ 39, pp 153-162.

[Prerau 75] D. S. PRERAU: DO - RE - MI: A Program that Recognizes Music Notation. In Computers and the Humanities 9, pp 25-29 (1975).

[Roads 85] C. ROADS: Research in Music and Artificial Intelligence. In ACM Computing Surveys, pp 162-186 (1985).

[Wanske 84] H. WANSKE: Musiknotation: Von der Syntax des Notenstichs zum EDV-gesteuerten Notensatz; Schott Musikwissenschaft, ISBN 3- 7957- 2886-x.

[Webster 91] B. F. WEBSTER: Making Music . In MacWorld, March 1991 pp 85-91.

[Wessel 86] D. WESSEL: MIDI-Lisp: A LISP-Based Music Programming Environment for the MacIntosh. In Proccedings of the International Computer Music Conference 1986, The Hague, pp 183-186.

[Wong 82] K.Y WONG, R.G. CASEY und F.M WAHL: Document analysis system, IBM J. Research Development 26, No. 6 (1982), pp 647-656.

[Ziegenrücker 88] W. ZIEGENRÜCKER: Allgemeine Musiklehre; Serie Musik Piper-Schott, ISBN 3 - 7957 - 8201 - 5 (Schott).

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI
-Bibliothek-
PF 2080
67608
Kaiserslautern
FRG

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen sowie die aktuelle Liste von allen bisher erschienenen Publikationen können von der oben angegebenen Adresse oder per anonymem ftp von ftp.dfki.uni-kl.de (131.246.241.100) unter pub/Publications bezogen werden.
Die Berichte werden, wenn nicht anders gekennzeichnet, kostenlos abgegeben.

DFKI Publications

The following DFKI publications or the list of all published papers so far are obtainable from the above address or via anonymous ftp from ftp.dfki.uni-kl.de (131.246.241.100) under pub/Publications.
The reports are distributed free of charge except if otherwise indicated.

DFKI Research Reports

RR-93-04
Christoph Klauck, Johannes Schwagereit:
GGD: Graph Grammar Developer for features in CAD/CAM
13 pages

RR-93-05
Franz Baader, Klaus Schulz: Combination Techniques and Decision Problems for Disunification
29 pages

RR-93-06
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: On Skolemization in Constrained Logics
40 pages

RR-93-07
Hans-Jürgen Bürckert, Bernhard Hollunder, Armin Laux: Concept Logics with Function Symbols
36 pages

RR-93-08
Harold Boley, Philipp Hanschke, Knut Hinkelmann, Manfred Meyer: COLAB: A Hybrid Knowledge Representation and Compilation Laboratory
64 pages

RR-93-09
Philipp Hanschke, Jörg Würtz:
Satisfiability of the Smallest Binary Program
8 pages

RR-93-10
Martin Buchheit, Francesco M. Donini, Andrea Schaerf: Decidable Reasoning in Terminological Knowledge Representation Systems
35 pages

RR-93-11
Bernhard Nebel, Hans-Juergen Buerckert:
Reasoning about Temporal Relations:
A Maximal Tractable Subclass of Allen's Interval Algebra
28 pages

RR-93-12
Pierre Sablayrolles: A Two-Level Semantics for French Expressions of Motion
51 pages

RR-93-13
Franz Baader, Karl Schlechta:
A Semantics for Open Normal Defaults via a Modified Preferential Approach
25 pages

RR-93-14
Joachim Niehren, Andreas Podelski,Ralf Treinen:
Equational and Membership Constraints for Infinite Trees
33 pages

RR-93-15
Frank Berger, Thomas Fehrle, Kristof Klöckner, Volker Schölles, Markus A. Thies, Wolfgang Wahlster: PLUS - Plan-based User Support Final Project Report
33 pages

RR-93-16
Gert Smolka, Martin Henz, Jörg Würtz: Object-Oriented Concurrent Constraint Programming in Oz
17 pages

RR-93-17
Rolf Backofen:
Regular Path Expressions in Feature Logic
37 pages

RR-93-18
Klaus Schild: Terminological Cycles and the Propositional μ-Calculus
32 pages

RR-93-20
Franz Baader, Bernhard Hollunder:
Embedding Defaults into Terminological Knowledge Representation Formalisms
34 pages

RR-93-22
Manfred Meyer, Jörg Müller:
Weak Looking-Ahead and its Application in Computer-Aided Process Planning
17 pages

RR-93-23
Andreas Dengel, Ottmar Lutzy:
Comparative Study of Connectionist Simulators
20 pages

RR-93-24
Rainer Hoch, Andreas Dengel:
Document Highlighting —
Message Classification in Printed Business Letters
17 pages

RR-93-25
Klaus Fischer, Norbert Kuhn: A DAI Approach to Modeling the Transportation Domain
93 pages

RR-93-26
Jörg P. Müller, Markus Pischel: The Agent Architecture InteRRaP: Concept and Application
99 pages

RR-93-27
Hans-Ulrich Krieger:
Derivation Without Lexical Rules
33 pages

RR-93-28
Hans-Ulrich Krieger, John Nerbonne, Hannes Pirker: Feature-Based Allomorphy
8 pages

RR-93-29
Armin Laux: Representing Belief in Multi-Agent Worlds viaTerminological Logics
35 pages

RR-93-30
Stephen P. Spackman, Elizabeth A. Hinkelman:
Corporate Agents
14 pages

RR-93-31
Elizabeth A. Hinkelman, Stephen P. Spackman:
Abductive Speech Act Recognition, Corporate Agents and the COSMA System
34 pages

RR-93-32
David R. Traum, Elizabeth A. Hinkelman:
Conversation Acts in Task-Oriented Spoken Dialogue
28 pages

RR-93-33
Bernhard Nebel, Jana Koehler:
Plan Reuse versus Plan Generation: A Theoretical and Empirical Analysis
33 pages

RR-93-34
Wolfgang Wahlster:
Verbmobil Translation of Face-To-Face Dialogs
10 pages

RR-93-35
Harold Boley, François Bry, Ulrich Geske (Eds.):
Neuere Entwicklungen der deklarativen KI-Programmierung — *Proceedings*
150 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

RR-93-36
Michael M. Richter, Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Ralf Legleitner, Gabriele Schmidt: Von IDA bis IMCOD: Expertensysteme im CIM-Umfeld
13 Seiten

RR-93-38
Stephan Baumann: Document Recognition of Printed Scores and Transformation into MIDI
24 pages

RR-93-40
Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt, Andrea Schaerf:
Queries, Rules and Definitions as Epistemic Statements in Concept Languages
23 pages

RR-93-41
Winfried H. Graf: LAYLAB: A Constraint-Based Layout Manager for Multimedia Presentations
9 pages

RR-93-42
Hubert Comon, Ralf Treinen:
The First-Order Theory of Lexicographic Path Orderings is Undecidable
9 pages

RR-93-43
M. Bauer, G. Paul: Logic-based Plan Recognition for Intelligent Help Systems
15 pages

RR-93-44
Martin Buchheit, Manfred A. Jeusfeld, Werner Nutt, Martin Staudt: Subsumption between Queries to Object-Oriented Databases
36 pages

RR-93-45
Rainer Hoch: On Virtual Partitioning of Large Dictionaries for Contextual Post-Processing to Improve Character Recognition
21 pages

RR-93-46
Philipp Hanschke: A Declarative Integration of Terminological, Constraint-based, Data-driven, and Goal-directed Reasoning
81 pages

RR-93-48
Franz Baader, Martin Buchheit, Bernhard Hollunder: Cardinality Restrictions on Concepts
20 pages

RR-94-01
Elisabeth André, Thomas Rist:
Multimedia Presentations:
The Support of Passive and Active Viewing
15 pages

RR-94-02
Elisabeth André, Thomas Rist:
Von Textgeneratoren zu Intellimedia-Präsentationssystemen
22 pages

RR-94-03
Gert Smolka:
A Calculus for Higher-Order Concurrent Constraint Programming with Deep Guards
34 pages

RR-94-05
Franz Schmalhofer,
J.Stuart Aitken, Lyle E. Bourne jr.:
Beyond the Knowledge Level: Descriptions of Rational Behavior for Sharing and Reuse
81 pages

RR-94-07
Harold Boley: Finite Domains and Exclusions as First-Class Citizens
25 pages

RR-94-08
Otto Kühn, Björn Höfling: Conserving Corporate Knowledge for Crankshaft Design
17 pages

DFKI Technical Memos

TM-92-01
Lijuan Zhang: Entwurf und Implementierung eines Compilers zur Transformation von Werkstückrepräsentationen
34 Seiten

TM-92-02
Achim Schupeta: Organizing Communication and Introspection in a Multi-Agent Blocksworld
32 pages

TM-92-03
Mona Singh:
A Cognitiv Analysis of Event Structure
21 pages

TM-92-04
Jürgen Müller, Jörg Müller, Markus Pischel, Ralf Scheidhauer:
On the Representation of Temporal Knowledge
61 pages

TM-92-05
Franz Schmalhofer, Christoph Globig, Jörg Thoben:
The refitting of plans by a human expert
10 pages

TM-92-06
Otto Kühn, Franz Schmalhofer: Hierarchical skeletal plan refinement: Task- and inference structures
14 pages

TM-92-08
Anne Kilger: Realization of Tree Adjoining Grammars with Unification
27 pages

TM-93-01
Otto Kühn, Andreas Birk: Reconstructive Integrated Explanation of Lathe Production Plans
20 pages

TM-93-02
Pierre Sablayrolles, Achim Schupeta:
Conlfict Resolving Negotiation for COoperative Schedule Management
21 pages

TM-93-03
Harold Boley, Ulrich Buhrmann, Christof Kremer:
Konzeption einer deklarativen Wissensbasis über recyclingrelevante Materialien
11 pages

TM-93-04
Hans-Günther Hein: Propagation Techniques in WAM-based Architectures — The FIDO-III Approach
105 pages

TM-93-05
Michael Sintek: Indexing PROLOG Procedures into DAGs by Heuristic Classification
64 pages

DFKI Documents

D-93-03
Stephan Busemann, Karin Harbusch(Eds.):
DFKI Workshop on Natural Language Systems: Reusability and Modularity - Proceedings
74 pages

D-93-04
DFKI Wissenschaftlich-Technischer Jahresbericht 1992
194 Seiten

D-93-05
Elisabeth André, Winfried Graf, Jochen Heinsohn, Bernhard Nebel, Hans-Jürgen Profitlich, Thomas Rist, Wolfgang Wahlster:
PPP: Personalized Plan-Based Presenter
70 pages

D-93-06
Jürgen Müller (Hrsg.):
Beiträge zum Gründungsworkshop der Fachgruppe Verteilte Künstliche Intelligenz, Saarbrücken, 29. - 30. April 1993
235 Seiten
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-07
Klaus-Peter Gores, Rainer Bleisinger:
Ein erwartungsgesteuerter Koordinator zur partiellen Textanalyse
53 Seiten

D-93-08
Thomas Kieninger, Rainer Hoch:
Ein Generator mit Anfragesystem für strukturierte Wörterbücher zur Unterstützung von Texterkennung und Textanalyse
125 Seiten

D-93-09
Hans-Ulrich Krieger, Ulrich Schäfer:
TDL ExtraLight User's Guide
35 pages

D-93-10
Elizabeth Hinkelman, Markus Vonerden,Christoph Jung: Natural Language Software Registry
(Second Edition)
174 pages

D-93-11
Knut Hinkelmann, Armin Laux (Eds.):
DFKI Workshop on Knowledge Representation Techniques — Proceedings
88 pages

D-93-12
Harold Boley, Klaus Elsbernd, Michael Herfert, Michael Sintek, Werner Stein:
RELFUN Guide: Programming with Relations and Functions Made Easy
86 pages

D-93-14
Manfred Meyer (Ed.): Constraint Processing – Proceedings of the International Workshop at CSAM'93, July 20-21, 1993
264 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-15
Robert Laux: Untersuchung maschineller Lernverfahren und heuristischer Methoden im Hinblick auf deren Kombination zur Unterstützung eines Chart-Parsers
86 Seiten

D-93-16
Bernd Bachmann, Ansgar Bernardi, Christoph Klauck, Gabriele Schmidt: Design & KI
74 Seiten

D-93-20
Bernhard Herbig:
Eine homogene Implementierungsebene für einen hybriden Wissensrepräsentationsformalismus
97 Seiten

D-93-21
Dennis Drollinger:
Intelligentes Backtracking in Inferenzsystemen am Beispiel Terminologischer Logiken
53 Seiten

D-93-22
Andreas Abecker: Implementierung graphischer Benutzungsoberflächen mit Tcl/Tk und Common Lisp
44 Seiten

D-93-24
Brigitte Krenn, Martin Volk:
DiTo-Datenbank: Datendokumentation zu Funktionsverbgefügen und Relativsätzen
66 Seiten

D-93-25
Hans-Jürgen Bürckert, Werner Nutt (Eds.):
Modeling Epistemic Propositions
118 pages
Note: This document is available only for a nominal charge of 25 DM (or 15 US-$).

D-93-26
Frank Peters:
Unterstützung des Experten bei der Formalisierung von Textwissen
INFOCOM - Eine interaktive Formalisierungskomponente
58 Seiten

D-94-01
Josua Boon (Ed.):
DFKI-Publications: The First Four Years 1990 - 1993
75 pages